Research Paper | Computer Science | E-ISSN No : 2454-9916 | Volume : 6 | Issue : 9 | Sep 2020

PATH PLANNING FOR DRONES

Sanket Kale[1] | Pravin Kadam[1] | Varad Chiney, Nikhil Joshi[1] | Prof. Harmeet Khanuja[1]

[1]Computer Department, MMCOE, Pune, India.

ABSTRACT

Path Planning for autonomous robots is finding the shortest optimum path from a given source to a destination while also taking obstacles into consideration. In our case the autonomous robots are drones. The algorithm needs to work dynamically and continuously as the drone is traversing as all the information about the environment would not be given initially but will be perceived as traversing. Thus the purpose of this project is to study and compare 3 methods for path planning on drones with low-end hardware, thus the algorithm needs to have time and space complexity less than or equal to what is required. The algorithms that are studied are path planning using fuzzy logic, path planning using reinforcement learning, and modified A-star algorithm.

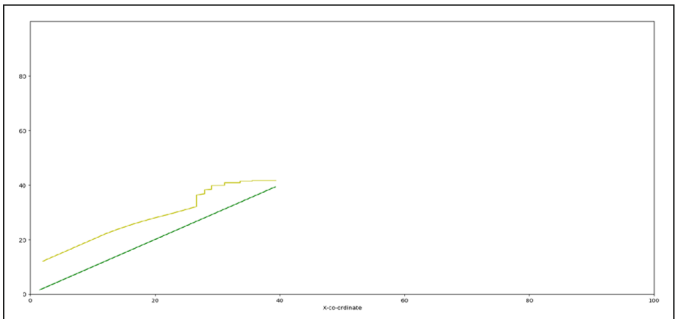
INTRODUCTION

Path Planning for autonomous robots is finding the shortest optimum path from a given source to a destination while also taking obstacles into consideration. In this study, we implemented 3 algorithms and compare the result to get the algorithm most suitable for the low end hardware drone. The algorithms that are studied are path planning using fuzzy logic, path planning using reinforcement learning, and modified A-star algorithm.

THEORY:

Path Planning using Fuzzy Logic:

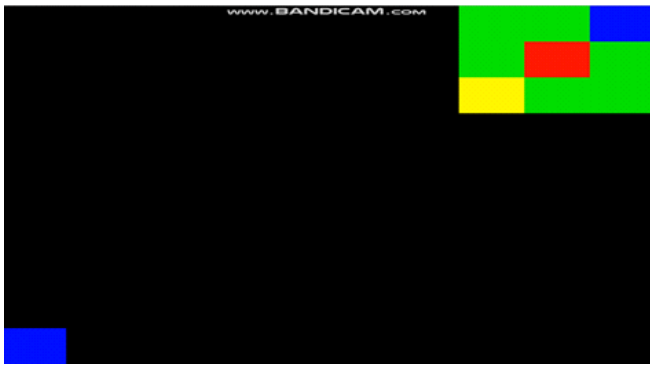
Fuzzy Logic is a method that resembles human reasoning. The approach imitates human reasoning as in considers intermediate possibilities, conventional computer understands precise input like binary input or integer input and gives precise output like true or false. This was one of the approaches used for making a path planning algorithm. We built an algorithm a for 2-dimensional case to test how well the algorithm worked. The algorithm took in velocity, distance from the destination as input, and gives out the amount of distance to traverse in x, y coordinates as output. Different parameters were taken into consideration like the maximum velocity allowed, the maximum allowed acceleration in unit time, the minimum distance from an obstacle before it acts considers it one. In case of multiple obstacles around, it only considered one obstacle, the one which was closest to the drone, this was one of its limitations. The output considered the amount to be acceleration and the amount of deceleration/acceleration in case of any obstacle. The inference engine for planning the path without any obstacle was very easy but as soon as obstacle came into play in a 2-dimensional case the complication increased. The number of rules required to take into consideration each case became exponential as compared to before and thus complexity for making the algorithm efficient increased as well. It is also considered only one of the obstacles. Considering how many cases were required for a 2-dimensional problem, in a 3-dimensional case, it would be even more. This approach had its advantages like the reliability was definite unlike the algorithm explained further as well as it took into account the velocity of the drone but it was too complex to generate cases for the inference engine, we needed an approach which was generalized and did not require specific cases and rules to be put into for it to work as expected.



Fuzzy Logic Output

Path Planning using Reinforcement Learning :

Reinforcement learning is an area of machine learning concerned with how software agents ought to take actions in an environment to maximize the notion of cumulative reward. So we used reinforcement learning to create a path planning algorithm. The approach was the simple reward the system as the distance between the destination and current position decreases. The inputs were the distance between source and destination and the output was move unit distance in one 8 directions as in north, south, east, west,north-west, north-east, south-west and south-east. Any obstacle with a range was also taken into consideration for example within 1 unit range, then all objects in the 8 directions. Though again a limited number of objects can be considered. After training the model, it worked but only for about 85% of the times and it did not necessarily take the optimum or the shortest path. Sometimes it even got stuck in one place and just kept moving in the vicinity. It was not efficient nor reliable. Even with extensive hyper parameter tuning, it did not work as desired. Thus, even though this algorithm had its advantages like simplicity and constant time complexity (about the time required to process throug the neural network), we had to discard this one as well.



Reinforcement Learning Output

Path Planning using a modified version of the A-star algorithm:

The A-star algorithm is a searching algorithm that searches for the shortest path between the source and destination point. The biggest problem with the A-star algorithm is the exponential time complexity. It has that big of a time complexity because of the search space it is required to search the optimum path. So to solve this problem we designed A star algorithm with custom search space as in the search space is reduced and a temporary destination is selected which will get us closest to the actual destination. So it keeps going towards the temporary destination until the actual destination is within range. The problem of time complexity is solved by reducing search space and it considers as many obstacles as the search space can fit. It finds the shortest path every single time. The algorithm runs after every unit time, thus dynamic obstacles are also taken into consideration.

Application:

The path planning algorithm can be used for any autonomous robot which is used for various applications.

CONCLUSION

Thus we have studied three path planning algorithms, built them from scratch, and found the algorithm that could be used on the drone. Out of the three a-star algorithm was the most efficient and reliable algorithm.

REFERENCES:

I. Bart Remes, Dino Hensen, Freek van Tienen, Christophe De Wagter, Erik van der Horst, and Guido de Croon, "Paparazzi: how to make a swarm of Parrot AR Drones fly autonomously based on GPS", published in the International Micro Air Vehicle Conference and Flight Competition (IMAV2013) 17-20 September 2013, Toulouse, France

II. Axel Bürkle, Matthias Kollmann,Florian Sego," Towards Autonomous Micro UAV Swarms", published in Journal of Intelligent & Robotic Systems on 27 October 2010

III. Jose Luis Sanchez-Lopez ; Jesús Pestana ; Paloma de la Puente ; Ramon Suarez-Fernandez, "A system for the design and development of vision-based multi-robot quadrotor swarms" published in IEEE in 2014 May from USA